

2. Rejection of Claims 1-20 Under 35 U.S.C. 101

Claims 1-20 stand rejected under 35 U.S.C. 101 as being directed to non-statutory subject matter. The Examiner indicates that the apparatus/computer-based method of claims 1-20 do not produce a useful, concrete, and tangible result, and that it is unclear how the result is being stored, displayed, or used in any tangible manner. Furthermore, the Examiner invites the Applicants to view the ***new guidelines for 35 U.S.C. 101*** (emphasis added) at <http://www.uspto.gov/web/offices/com/sol/og/2005/week47/patgupa.htm>. Applicants traverse this rejection of claims 1-20 under 35 U.S.C. 101. Reconsideration is respectfully requested.

Applicants have reviewed MPEP 2106, which indicates:

The claimed invention as a whole must accomplish a practical application. That is, it must produce a "useful, concrete and tangible result." State Street, 149 F.3d at 1373, 47 USPQ2d at 1601-02. The purpose of this requirement is to limit patent protection to inventions that possess a certain level of "real world" value...

Apart from the utility requirement of 35 U.S.C. 101, usefulness under the patent eligibility standard requires significant functionality to be present to satisfy the useful result aspect of the practical application requirement. See Arrhythmia, 958 F.2d at 1057, 22 USPQ2d at 1036. Merely claiming nonfunctional descriptive material stored in a computer-readable medium does not make the invention eligible for patenting. For example, a claim directed to a word processing file stored on a disk may satisfy the utility requirement of 35 U.S.C. 101 since the information stored may have some "real world" value. However, the mere fact that the claim may satisfy the utility requirement of 35 U.S.C. 101 does not mean that a useful result is achieved under the practical application requirement. The claimed invention as a whole must produce a "useful, concrete and tangible" result to have a practical application.

Although the courts have yet to define the terms useful, concrete, and tangible in the context of the practical application requirement for purposes of these guidelines, the following examples illustrate claimed inventions that have a practical application because they produce useful, concrete, and tangible result:

- Claims drawn to a ***long-distance telephone billing process*** containing mathematical algorithms were held to be directed to patentable subject matter because "the claimed process applies the Boolean principle to produce a useful, concrete, tangible result without pre-empting other uses of the mathematical

principle." AT&T Corp. v. Excel Communications, Inc., 172 F.3d 1352, 1358, 50 USPQ2d 1447, 1452 (Fed. Cir. 1999);

- "[T]ransformation of data, representing discrete dollar amounts, by a machine through **a series of mathematical calculations into a final share price**, constitutes a practical application of a mathematical algorithm, formula, or calculation, because it produces 'a useful, concrete and tangible result' -- a final share price momentarily fixed for recording and reporting purposes and even accepted and relied upon by regulatory authorities and in subsequent trades." State Street, 149 F.3d at 1373, 47 USPQ2d at 1601; and

- Claims drawn to a rasterizer for converting discrete waveform data samples into **anti-aliased pixel illumination intensity data to be displayed on a display means** were held to be directed to patentable subject matter since the claims defined "a specific machine to produce a useful, concrete, and tangible result." In re Alappat, 33 F.3d 1526, 1544, 31 USPQ2d 1545, 1557 (Fed. Cir. 1994). (Emphasis added.)

Applicants assert that independent claims 1, 14, and 19 include practical applications that produce useful, concrete, and tangible results.

Independent claim 1 calls for **apparatus comprising computer readable media**; and program code, stored on the computer readable media, comprising code to define a user interface; **code to detect invalid test definition data in user input and, upon detection of invalid test definition data, prompt a user to select a valid data option from a set of valid data options**; said prompting being undertaken through the user interface; and code to receive a valid data option selected through the user interface, and to update the invalid test definition data with the valid data option. Applicants assert that the present invention, which includes apparatus having computer readable media together with code to prompt a user to select a valid data option from a set of valid data options, does in fact produce a useful, concrete, and tangible result. Accordingly, independent claim 1 is believed to be statutory.

Claims 2-13, which each depend either directly or ultimately from independent claim 1, are believed to be statutory for at least the above-identified reasons.

Independent claim 14 calls for **a computer-based method** comprising parsing user input to detect invalid test definition data in the user input; upon detecting invalid test definition data, **prompting a user to select a valid data option from a set of valid data options**; upon receiving a valid data option selected from the set of valid

data options, ***updating the invalid test definition data with the valid data option***; and ***generating circuit test data structures to control an automated circuit tester***. Applicants assert that the present invention, which includes a computer-based method prompting a user to select a valid data option from a set of valid data options, updating invalid test definition data with the valid data option, and generating circuit test data structures to control an automated circuit tester, does in fact produce a useful, concrete, and tangible result. Accordingly, independent claim 14 is believed to be statutory.

Claims 15-18, which each depend directly from independent claim 1, are believed to be statutory for at least the above-identified reasons.

Independent claim 19 calls for ***a computer-based method comprising*** parsing source code for generating circuit test data structures, to identify type name definitions and enumeration constant definitions contained in said source code; generating a string table from said type name and enumeration constant definitions; and ***linking said string table to an input validation and error messaging portion of said source code to i) cause said source code to index said string table upon detection of invalid test definition data in user input, and then ii) cause a set of valid data options retrieved from said string table to be displayed to a user for user selection***. Applicants assert that the present invention, which includes a computer-based method linking a string table to an input validation and error messaging portion of source code to cause a set of valid data options retrieved from said string table to be displayed to a user for user selection, does in fact produce a useful, concrete, and tangible result. Accordingly, independent claim 19 is believed to be statutory.

Claim 20, which depends directly from independent claim 1, is believed to be statutory for at least the above-identified reasons.

3. Rejection of Claims 1-7, 10-16, 19 and 20 Under 35 U.S.C. 103(a)

Claims 1-7, 10-16, 19 and 20 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Colby et al. (U.S. Patent No. 6,622,271; referred to herein as

"Colby") and further in view of Gygi et al. (U.S. Patent Application Publication No. 2003/0235156 A1; referred to herein as "Gygi"). Applicants traverse this rejection for the following reasons. Reconsideration is respectfully requested.

Independent claim 1 calls for **apparatus comprising** computer readable media; and program code, stored on the computer readable media, comprising code to define a user interface; **code to detect invalid test definition data in user input and, upon detection of invalid test definition data, prompt a user to select a valid data option from a set of valid data options**; said prompting being undertaken through the user interface; and code to receive a valid data option selected through the user interface, and to update the invalid test definition data with the valid data option.

With respect to applicants' claim 1, the Examiner asserts in the Office Action of 6/15/2006 at page 2 (underlining added) that Colby:

disclose[s] an apparatus, comprising...code to receive a valid data option selected through the user interface, and to update the invalid test definition data with valid data option (col. 11, lines 52-57).

Applicants respectfully disagree. See, for example, Colby at col. 11, lines 52-57:

If a problem is detected, then a warning message is provided to the operator, so that appropriate adjustments can be made to the test definition 73. After any appropriate adjustments are made, the resulting test definition 73 is stored in each of the testers 17 and 18, in particular at 107 and 207, respectively.

Applicants assert that Colby, *arguendo*, may suggest apparatus having code to receive a valid data option **provided** through a user interface, and to update invalid test definition data with the valid data option. However, Colby does not teach or suggest apparatus having code to receive a valid data option **selected** through a user interface, and to update invalid test definition data with the valid data option. In other words, Colby does not teach or suggest a valid data option taken as a choice

from among several valid data options. Colby at col. 12, lines 20-29 further discloses (emphasis added):

If any errors are detected, the interpreter program 131 will provide the operator with an identification of those errors. ***The operator has the capability to carry out certain debug functions of a standard type, such as*** setting breakpoints, ***dynamically changing the values of variables***, and so forth. Further, the operator can instruct the interpreter program 131 to make changes to the modified test definition 301, for example to correct errors which were present in the initial test definition 73, or to implement special test conditions to help identify an elusive problem in a particular device 12.

Furthermore, the Examiner previously asserted that Colby teaches “code to...prompt a user to select a valid data option from a set of valid data options. . . (col. 4, lines 54-67 to col. 5, lines 1-4; col. 11, lines 45-55; col. 12, lines 20-29)”. See, 6/24/2005 Office Action, p. 1. Now, the Examiner asserts that Colby does not teach “upon detection of invalid test definition data, prompt a user to select a valid data option from a set of valid data option, said prompting being undertaken through the user interface...”

On page 4 of the Office Action of 06/14/2006, the Examiner states:

Gygi et al. disclose an apparatus comprising...code to detect invalid test definition data in user input and, upon detection of invalid test definition data, prompt a user to select a valid data option from a set of valid data option, said prompting being undertaken through the user interface, code to compile the set of valid data options based on a context of the invalid test definition data to index a table of valid data options, to parse the user input and log valid data options into the table, wherein the context comprises a data type, the code to configure how the set of valid data options is displayed through the user interface, and the set of valid data options comprises a single valid data option that is replaceable by the user, and cause a set of valid data options retrieved from the string table to be displayed to a user for user selection ([0048], [0050], [0051], [0068], and [0069]).

Applicants respectfully disagree. Gygi discloses a flexible command, status and parameter definition language that permits a test designer to define a wide variety of custom commands, test specific status and test parameters to be supplied by the test operator in starting a selected test vehicle. Gygi discloses definitions including types

and ranges of permissible values as well as user interface information to prompt the test operator for desired values. Gygi discloses default values for the parameter to assume if the operator does not specify a value for the parameter when prompted to do so and may provide textual help messages to describe the usage of the parameter or permissible values for the parameter. Although Colby and Gygi enables a user to dynamically change the value of an incorrect variable, neither suggest that a user is **prompted to select a valid data option**, as is set forth in Applicants' claim 1.

Prompting a user to select a valid data option is not only novel and unobvious over the teachings of Colby and Gygi, but it is especially useful because, as noted in applicants' specification, "Discerning the type of data that must be supplied to "cure" [an] error can be a laborious and time-consuming task." See, Applicants' specification, p. 2, lines 1-2. Accordingly, independent claim 1 is believed to allowable.

Claims 2-13, which each depend either directly or ultimately from independent claim 1, are believed to be allowable for at least the above-identified reasons.

Independent claim 14 calls for **a computer-based method** comprising parsing user input to detect invalid test definition data in the user input; upon detecting invalid test definition data, **prompting a user to select a valid data option from a set of valid data options**; upon receiving a valid data option selected from the set of valid data options, **updating the invalid test definition data with the valid data option**; and generating circuit test data structures to control an automated circuit tester. Applicants' claim 14 is believed to be allowable for reasons similar to why claim 1 is believed to be allowable.

Claims 15-18, which each depend directly from independent claim 1, are believed to be allowable for at least the above-identified reasons.

Independent claim 19 calls for **a computer-based method comprising** parsing source code for generating circuit test data structures, to identify type name definitions and enumeration constant definitions contained in said source code; generating a string table from said type name and enumeration constant definitions; and **linking said string table to an input validation and error messaging portion of said source code to i) cause said source code to index said string table upon**

detection of invalid test definition data in user input, and then ii) cause a set of valid data options retrieved from said string table to be displayed to a user for user selection.

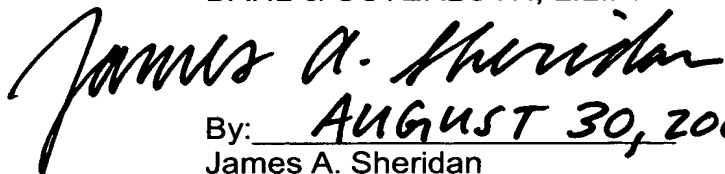
Applicants' claim 19 is believed to be allowable at least for the reason that neither Colby nor Gygi teach "...i) caus[ing] said source code to index said string table upon detection of invalid test definition data in user input, and then ii) caus[ing] a set of valid data options retrieved from said string table *to be displayed to a user for user selection.*" Similar to Colby and Gygi failing to teach "prompting" a user with valid data options, Colby and Gygi also fail to "display" valid data options for user selection. Accordingly, claim 19 is believed to be allowable.

Claim 20, which depends directly from independent claim 1, is believed to be allowable for at least the above-identified reasons.

Conclusion

In light of the amendments and remarks provided herein, Applicants respectfully request the timely issuance of a Notice of Allowance.

Respectfully submitted,
DAHL & OSTERLOTH, L.L.P.



By: AUGUST 30, 2006
James A. Sheridan
Reg. No. 43,114
Tel: (303) 291-3200